Detecting Novel Fault Conditions with Hidden Markov Models and Neural Networks

Padhraic Smyth
Jet Propulsion laboratory 238-420
California institute of Technology
4800 Oak Grove Drive
Pasadena, CA 91109
email: *pjs@galway.jpl. nasa.gov*
tel: (818) 3543768, fax: (818) 3546825.

1 Introduction

Conventional control-theoretic models for fault detection rely on an accurate model of the system being monitored: frequently in practice no such model exists for complex non-linear systems. The large ground antennas used by the Jet Propulsion Laboratory's Deep Space Network (DSN) to track planetary spacecraft fall into this category --- quite complicated analytical models exist for the electro-mechanical pointing systems, but they arc known to bc a poor fit for fault detection purposes.

We have previously described the application of online adaptive pattern recognition methods to this problem [1, 2]. The system operates as follows. Sensor data such as motor current, position encoder, tachometer voltages, and so forth are synchronously sampled at 5011 z by a data acquisition system. The data is blocked off into disjoint windows (200 samples are used in practice) and various features (such as autoregressive coefficients estimated online) are extracted; let the feature vector bc $\underline{\theta}$.

The features $\underline{\theta}$ are fed into a classification model (every 4 seconds) which in turn provides posterior probability estimates of the m possible states of the system given the estimated features from that window, $p(\omega_i|\underline{\theta})$. ω_1 corresponds to normal conditions, the other ω_i's, $1 \leq i \leq m$, correspond to known fault conditions.

Finally, since the system has "memory" in the sense that it is more likely to remain in the current state than to change states, the posterior probabilities need to bc correlated over time. This is achieved by a standard first-order hidden Markov model (II MM) which models the temporal state dependence [2].

As described in [1, 2] the classifier portion of the mode] is trained using simulated hardware faults. The feed-forward neural network has been the model of choice for this application because of its discrimination ability, its posterior probability estimation properties [3, 4] and its relatively simple implementation in software. Also described in [2] at length is the design of the II MM transition matrix based on prior knowledge of system mean time between failure (MTBF) information and other specific knowledge of the system configuration. An important point is that the HMM transition parameters are not learned from data as in speech recognition — in a sense, these "designed" II M M models can bc considered temporal context networks of the Bayesian network family described by l'earl and others [5].

2 I.incitations of the Discriminative HMM Model

The model described above assumes that there arc m known mutually exclusive and exhaustive states (or "classes") of the system, $w_1, \ldots, \omega_m$. The mutually exclusive assumption is reasonable in many applications where multiple simultaneous failures arc highly unlikely. However, the exhaustive assumption is somewhat impractical. In particular, for fault detection in a complex system such as the antenna, there arc literally thousands of possible fault conditions which might occur. The probability of occurrence of any single condition is very small, but nonetheless there is a significant probability that at least one oft hese conditions will occur over some finite time. While the common faults can be directly modelled it is not practical to assign specific states to all the other minor faults which might occur.

As discussed in [I] and [6], discriminative models directly model the posterior probabilities of the classes given the feature data and they assume that the classes arc exhaustive. On the other hand, a *generative* model directly models the data likelihood $p(\underline{\theta}|\omega_i)$ and then determines posterior class probabilities by application of Bayes' rule (see Smyth [6] and Dawid ['i'] for further discussion). Examples of generative classifiers include parametric models such as Gaussian classifiers and memory-based methods such as kernel density estimators and near neighbour models. Generative models are by nature well suited to novelty detection. However, there is a trade-of because generative models typically arc doing more modelling than just searching for a decision boundary, they can be less efficient (than discriminant methods) in their usc of the data.

3 Hybrid Models

A practical approach is to use both a generative and discriminative classifier and add an extra "$m+1$ th" state to the model to cover "all other possible states" not accounted for by the known m states. Hence, the posterior estimates of the generative classifier arc conditioned on whether or not the data is thought to come from one of the m known classes.

Let the symbol $\omega_{\{1,\ldots,m\}}$ denote the event that the true system state is one of the known states, and let $p(\omega_{m+1}|\underline{\theta})$ be the posterior probability that the data is from an unknown state. 11 ence, one can estimate the true posterior probability of individual known states as

$$\hat{p}(\omega_i|\underline{\theta}) = p_d(\omega_i|\underline{\theta}, \omega_{\{1,\ldots,m\}})p(\omega_{\{1,\ldots,m\}}|\underline{\theta}) = p_d(\omega_i|\underline{\theta}, \omega_{\{1,\ldots,m\}})(1 - p(\omega_{m+1}|\underline{\theta})), \qquad 1 \leq i \leq m \tag{1}$$

where $p_d(\omega_i|\underline{\theta}, \omega_{\{1,\ldots,m\}})$ is the posterior probability estimate of state *i* as provided by a discriminative model.

The calculation of $p(\omega_{m+1}|\underline{\theta})$ in Equation (1) can be obtained via the usual application Bayes' rule if $p(\underline{\theta}|\omega_{m+1})$, $p(\omega_{m+1})$, and $p(\underline{\theta}|\omega_{\{1,\ldots,m\}})$ are known, since

$$p(\omega_{m+1}|\underline{\theta}) = \frac{p(\underline{\theta}|\omega_{m+1})p(\omega_{m+1})}{p(\underline{\theta}|\omega_{m+1})p(\omega_{m+1}) + p(\underline{\theta}|\omega_{\{1,\ldots,m\}})\sum_i^m p(\omega_i)} \tag{2}$$

In practice we use non-informative Bayesian priors for $p(\underline{\theta}|\omega_{m+1})$ (in Equation (2)) over a bounded space of feature values (details arc available in a technical report [8]), although this choice of a prior density or data of unknown origin is basically ill-posed. The stronger the constraints which can be placed on the features, the narrower the prior density, and the better the ability of the overall model to detect novelty. If wc only have very weak prior information, this will translate into a weaker criterion for accepting points which belong to the unknown category.

The term $p(\omega_{m+1})$ in Equation (2) must be chosen based on tbc designer's prior belief of how often the system will be in an unknown state -- a practical choice is that the system is at least as likely to be in an unknown failure state as any of the known failure states. The $p(\underline{\theta}|\omega_{\{1,\ldots,m\}})$ term in Equation (2) is provided directly by the generative model. 'Typically this can be a mixture of Gaussians or a kernel density estimate over all of the training data (ignoring class labels).

Integration of equations (1) and (2) into the hidden Markov model calculations is straightforward and will not be derived — the model now has an extra state, "unknown." The choice of transition probabilities between the unknown and other states is once again a matter of design choice. For the antenna application at least, many of the unknown states are believed to be relatively brief transient phenomena which last perhaps no longer than a fcw seconds: hence the Markov matrix is designed to reflect these beliefs since the expected duration of any state $d[\omega_i]$ (in units of sampling intervals) for a *first*-order Markov model must obey

$$d[\omega_i] = \frac{1}{1 - p_{ii}} \tag{3}$$

where p_{ii} is the self-transition probability of state w_i.

An alternative approach to novelty detection which does not require tbc use of prior densities was proposed by Dubuisson and Masson [9]. This approach uses a generative model directly for classification and

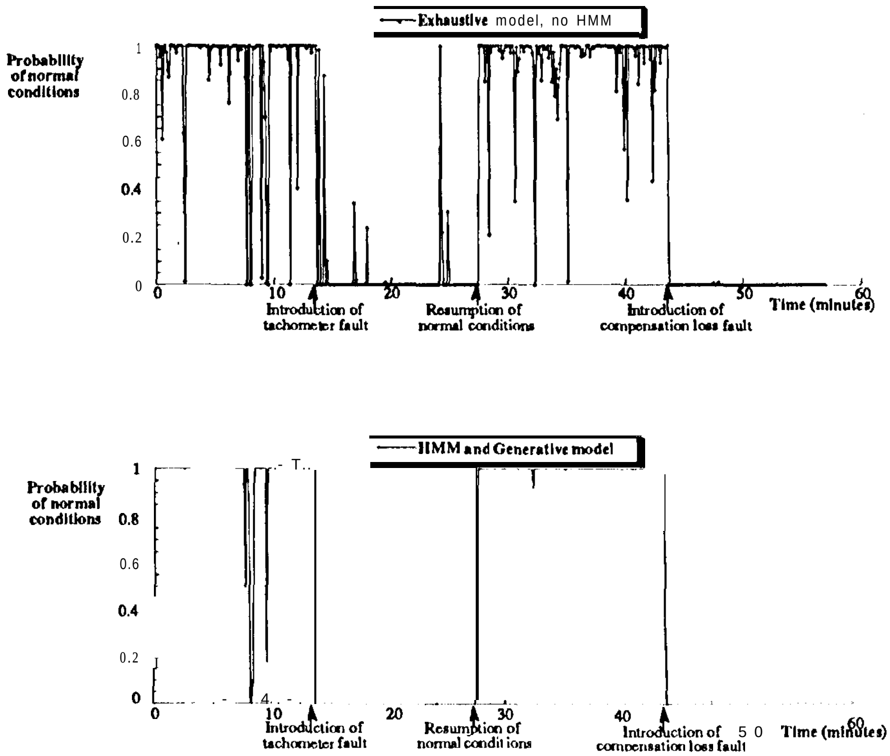

Figure 1: Estimated posterior probability of normal state (a) using no HMM and the exhaustive assumption (normal + 3 fault states), (b) using a HMM with a generative model (normal + 3 faults+ other state).

thus detects outliers via distance thresholds, e.g., the distance from the mean in a parametric model, or the distance from the nearest training set point in near-neighbour or kernel models. The disadvantage of threshold-based methods lies in the selection of the thresholds themselves; without some hypothesis for data from the unknown state there is no principled way to choose such thresholds.

4 Experimental Results

For comparison purposes we evaluated the results of 2 particular models. Each was applied to monitoring the servo pointing system of a particular DSN 34-meter antenna at Goldstone, California. The models were implemented within the LabView data acquisition package running in real-time on a Macintosh II at the antenna site. The models had previously been trained off-line on data collected some months earlier. 12 input features were used and the experiment consisted of introducing hardware faults into the system in a controlled manner at 15 minutes and 45 minutes, each of 15 minutes duration,

Figure 1 (a) and (b) show each model's estimates over time that the system is in the normal state (space limitations precluded the inclusion of more detailed experimental results). Model (a) uses no HMM and assumes that the 4 known states are exhaustive -- a single feedforward neural network with 8 hidden units was used as the discriminative model. Model (b) uses a HMM with 5 states, where a generative model (a Gaussian mixture model) and a flat prior (with bounds on the feature values) are used to determine the probability of the 5th state. The same neural network as in model (a) is used as a discriminator for the other 4 known states. The generative mixture model had 10 components and used only 2 of the 12 input features, the 2 which were judged to be the most sensitive to system change. The parameters of the HMM were designed according to the guidelines described earlier. Known fault states were assumed to be equally likely with 1 hour MTBF's and with 1 hour mean duration. Unknown faults were assumed to have a 20 minute MTBF and a 10 second mean duration.

Model (a)'s estimates are quite noisy and contain a significant number of potential false alarms (highly undesirable in an operational environment). Model (b) is much more stable due to the smoothing effect of the H MM. Nonetheless, we note that between the 8th and 10 minutes, there appear to be some possible false alarms: this data was classified into the unknown state (not shown). On later inspection it was found that large transients (of unknown origin) were in fact present in the original sensor data and that this was what the model had detected, confirming the result obtained independently by the classifier. It is worth pointing out that the model without a generative component (whether with or without the II MM) did in fact always detect a non-normal state at the same time, but incorrectly classified this state as one of the known fault states (these results are not shown).

5 Application Issues

The ability to detect previously unseen transient behaviour has important practical consequences: as well as being used to warn operators of antenna problems in real-time, the model can also be used as a filter to a data logger to record interesting and anomalous data on a continuous basis. Hence, potentially novel system characteristics can be recorded for correlation with other antenna-related events (such as maser problems, receiver lock drop during RF feedback tracking, etc.) for later analysis to uncover the true cause of the anomaly.

The basic model described in this abstract has recently been approved for inclusion as a functional requirement in the antenna controller design for all new DSN antennas. The first such antenna is currently being built at the Goldstone, California, DSN site and will become operational in 1994 --- similar antennas, also equipped with fault detectors of the type described here, will be constructed in Spain and Australia in the 1995-96 time-frame.

Acknowledgements

The research described in this paper was performed at the Jet l'repulsion laboratory, California institute of Technology, under a contract with the National Aeronautics and Space Administration and was supported in part by ARPA under grant number N00014-92-J-1860

References

1. P. Smyth and J. Mellstrom, 'Fault diagnosis of antenna pointing systems using hybrid neural networks and signal processing techniques,' in *Advances in Neural Information Processing Systems 4*, J. E. Moody, S. J. Hanson, R. P. Lippmann (eds.), Morgan Kaufmann Publishers: San Mateo, CA, 1992, pp.667-674.

2. P. Smyth, 'Hidden Markov models for fault detection in dynamic systems,' Pattern Recognition, to appear

3. M. D. Richard and R. P. Lippmann, 'Neural network classifiers estimate Bayesian a posteriori probabilities,' Neural Computation, 3(4), pp.461-483, 1992.

4. J. Miller, R. Goodman, and P. Smyth, 'On loss functions which minimize to conditional expected values and posterior probabilities,' *IEEE* Transactions on *Information* Theory, to appear, July 1993.

5. J. Pearl, Probabilistic reasoning in intelligent *systems*, Morgan Kaufmann Publishers: San Mateo, CA, 1988.

6. P. Smyth, 'Probability density estimation and local basis function neural networks,' in *Computational Learning Theory and Natural Learning* Systems: *Volume* .2, T. Petsche, M. Kearns, S. Hanson, R. Rivest (eds.), Cambridge, MA: MIT Press, 1993.

7. A. P. Dawid, 'Properties of diagnostic data distributions,' *Biometrics*, 32, pp.647-658, Sept. 1976.

8. P. Smyth and J. Mellstrom, 'Failure detection in dynamic systems: model construction without fault training data,' *Telecommuncations and Data Acquisition Progress Report, vol. 112*, Jet Propulsion Laboratory, Pasadena, CA, February 15th 1993.

9. B. Dubuisson and M. Masson, 'A statistical decision rule with incomplete knowledge about the classes,' *Pattern* Recognition, vol.26, no,], pp.155-165, 1993,